applying KMAN to KMAN (KCODE) to produce KCODE.

38. The medium of claim 37 wherein the security key of the processor is a symmetric key and the application is instantiated from a code image including a main body and a header including:

KCPU (KMAN)	KMAN encrypted according to KCPU
MAC (main body, KMAN)	message authentication code of the main body under KMAN
KMAN (KCODE)	KCODE encrypted according to KMAN

where KCPU is the security key, KMAN is a device key of the portable device independent of the security key, and KCODE is the secret of the application, and

wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises:

applying KCPU to KCPU (KMAN) to produce KMAN;

computing MAC (main body, KMAN);

comparing the computed MAC to MAC (main body, KMAN) from the header to determine if the code image has been changed; and

if the MACs match, applying KMAN to KMAN (KCODE) to produce KCODE.

39. The medium of claim 31 wherein the security key of the processor is a private key of a public key—private key pair and the application is instantiated from a code image including a main body and a header including:

public key (KCODE)	KCODE encrypted according to the public key

where KCODE is the secret of the application, and

wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises applying the security key as the private key to public key (KCODE) to produce KCODE.

40. The medium of claim 39 wherein the security key of the processor is a private key of a public key—private key pair and the application is instantiated from a code image including a main body and a header including:

public key (HASH (main body), KCODE)	Hash of the main body and KCODE, both encrypted according to the public key

where KCODE is the secret of the application, and

wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises:

computing HASH (main body);

applying the private key to public key (HASH (main body), KCODE) to produce HASH (main body) and KCODE;

comparing the computed HASH to the produced HASH to determine if the code image has been changed;; and

if the HASHs match, employing the produced KCODE as appropriate.

41. A computer-readable medium having computer-executable instructions thereon implementing a method for a secure processor to instantiate one of a plurality of available secure applications thereon by way of a security kernel, the method comprising:

setting a chooser value to a value corresponding to a chooser application upon power-up;

entering a preferred mode upon a power-up CPU reset and instantiating the security kernel, the security kernel determining that the chooser value corresponds to the chooser application and therefore authenticating same, the chooser application being instantiated;

entering a normal mode after the chooser application is instantiated and leaving same to run, the chooser application presenting the plurality of available applications for selection by a user;

receiving a selection of one of the presented applications to be instantiated;

setting the chooser value to a value corresponding to the selected application;

entering a preferred mode upon an executed CPU reset and instantiating the security kernel, the security kernel determining that the chooser value corresponds to the selected application and therefore authenticating same, the selected application being instantiated;

entering a normal mode after the selected application is instantiated and leaving same to run;

wherein the security kernel allows the processor to be trusted to keep hidden a secret of the chooser application and a secret of the selected application.

42. The medium of claim 41 wherein the method further comprises setting the chooser value to the value corresponding to the chooser application upon the selected application being authenticated by the security kernel, wherein upon execution of a CPU reset, the security kernel determines that the chooser value corresponds to the chooser application **72***c* and therefore authenticates same.

43. The medium of claim 41 wherein the method further comprises storing the chooser value in a memory location not affected by a CPU reset so that the stored chooser value is available after same.

44. A computer-readable medium having stored thereon computer-executable instructions implementing a method for a secure processor to instantiate a secure application thereon, the method comprising:

instantiating a first security kernel which employs symmetric cryptography;

instantiating by way of the instantiated first security kernel a second security kernel which employs asymmetric cryptography; and

authenticating by way of the instantiated second security kernel the secure application.

45. The medium of claim 44 wherein the security key of the processor is a symmetric key and the second security kernel is instantiated by the first security kernel from a code image including a main body and a header including:

KCPU (KMAN)	KMAN encrypted according to KCPU
KMAN (KCODE)	KCODE encrypted according to KMAN

where KCPU is a security key of the processor, KMAN is a device key independent of the security key, and KCODE is the private key of the second security kernel, and

wherein the first security kernel applies the security key to decrypt the private key of the second security kernel during instantiation thereof by:

applying KCPU to KCPU (KMAN) to produce KMAN; and

applying KMAN to KMAN (KCODE) to produce KCODE.

46. The medium of claim 45 wherein the application is instantiated by the second security kernel from a code image including a main body and a header including:

public key (KCODE)	KCODE encrypted according to the public key

where KCODE is the secret of the application, and

wherein the second security kernel applies the private key to decrypt the secret of the application during instantiation thereof.

* * * * *